# FIG. 1

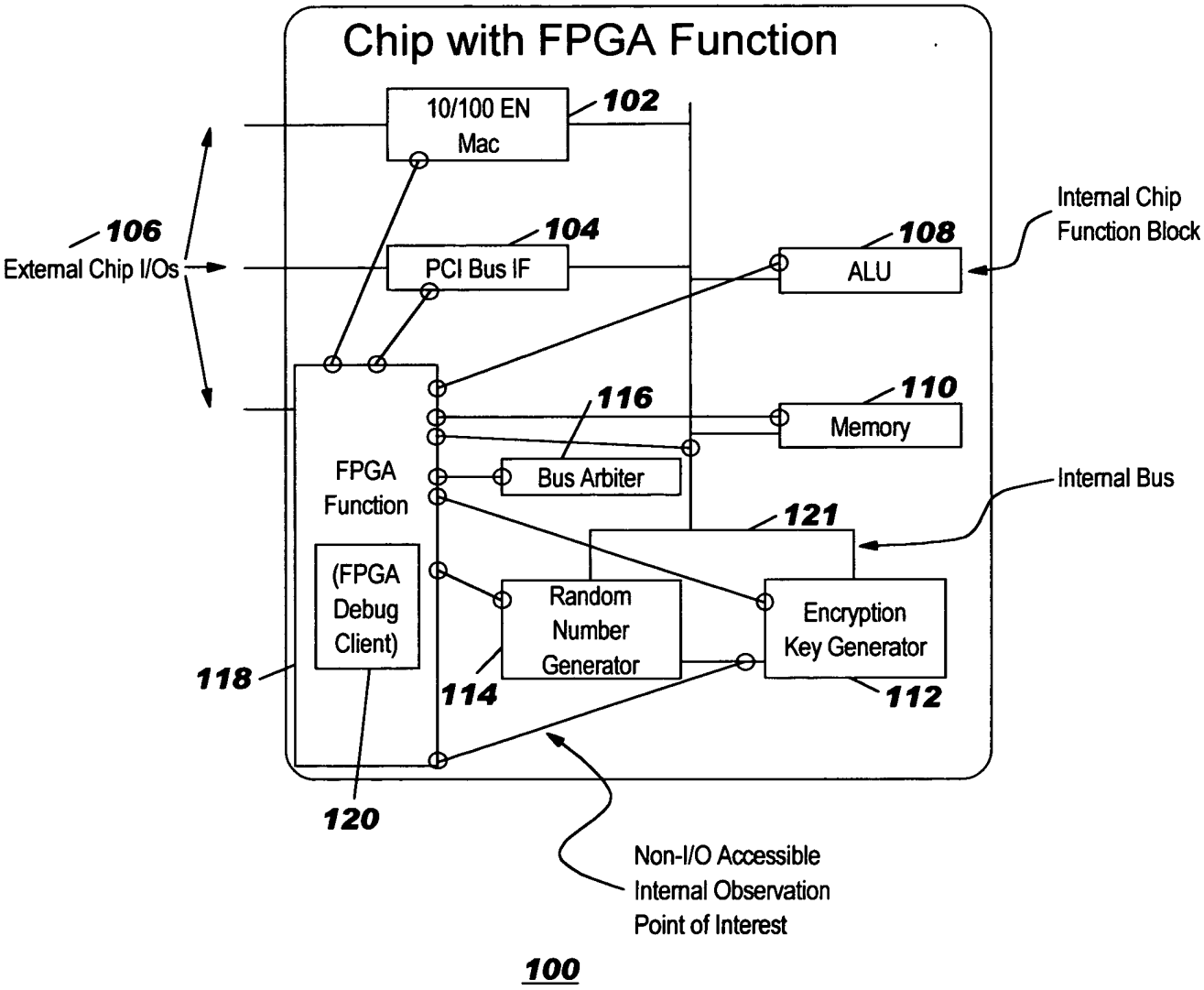
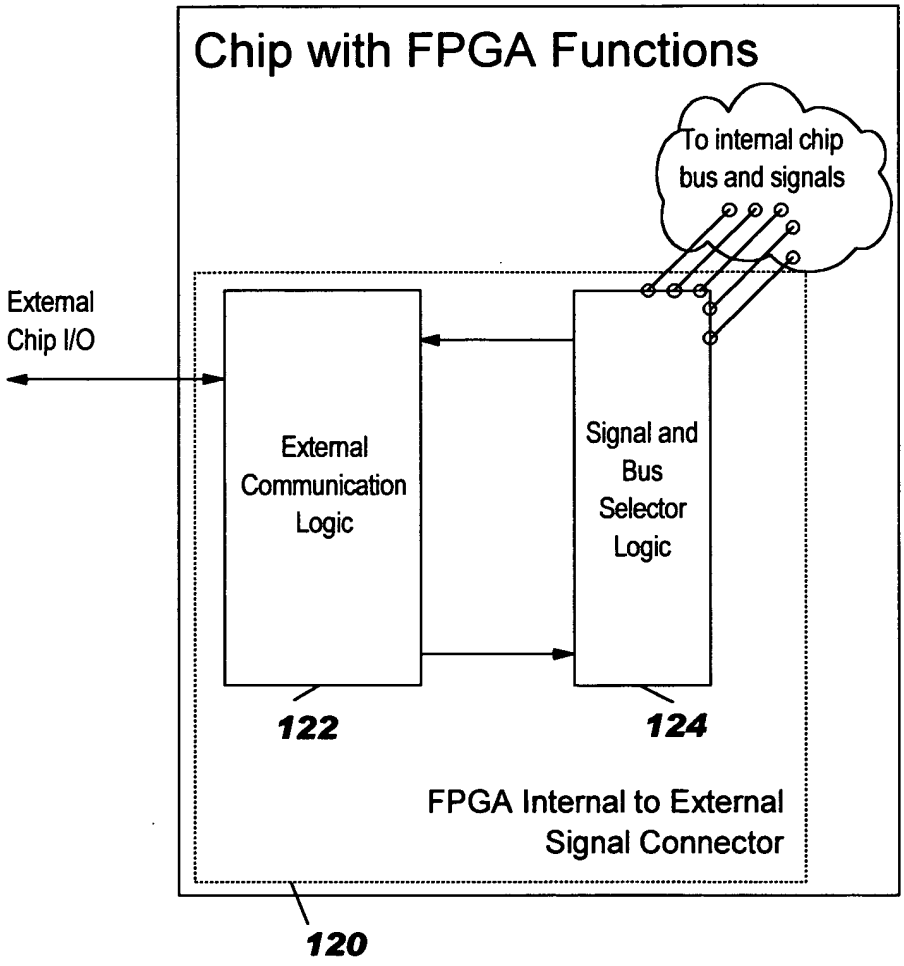
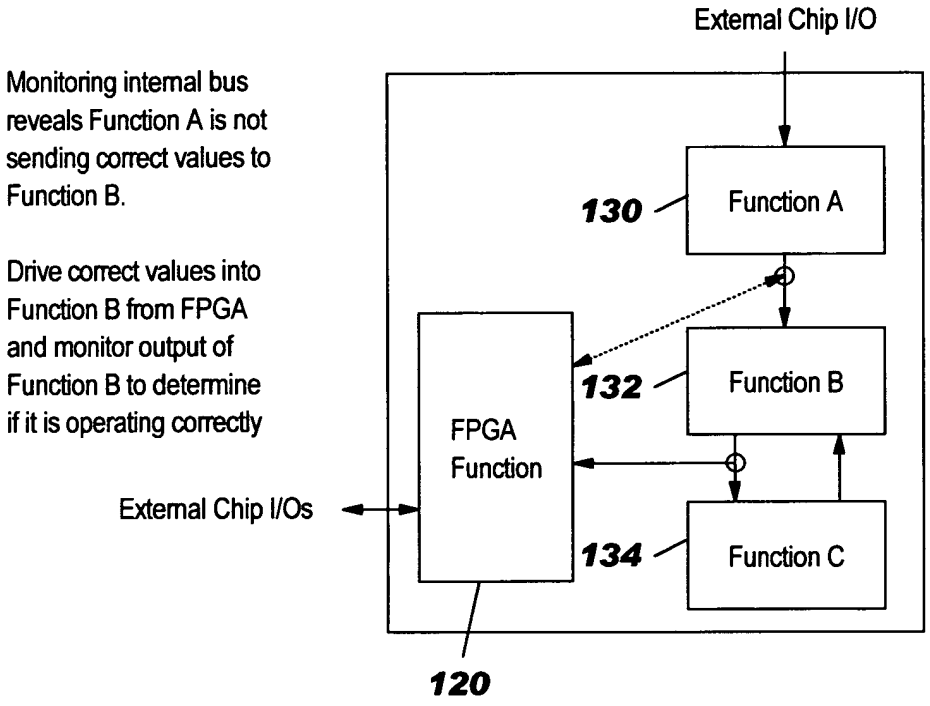

Chip with FPGA Function
10/100 EN Mac
102
106
External Chip I/Os
104
PCI Bus IF
108
ALU
Internal Chip Function Block
116
110
Memory
FPGA Function
Bus Arbiter
Internal Bus
121
(FPGA Debug Client)
Random Number Generator
Encryption Key Generator
118
114
112
120
Non-I/O Accessible Internal Observation Point of Interest
100

FIG. 2

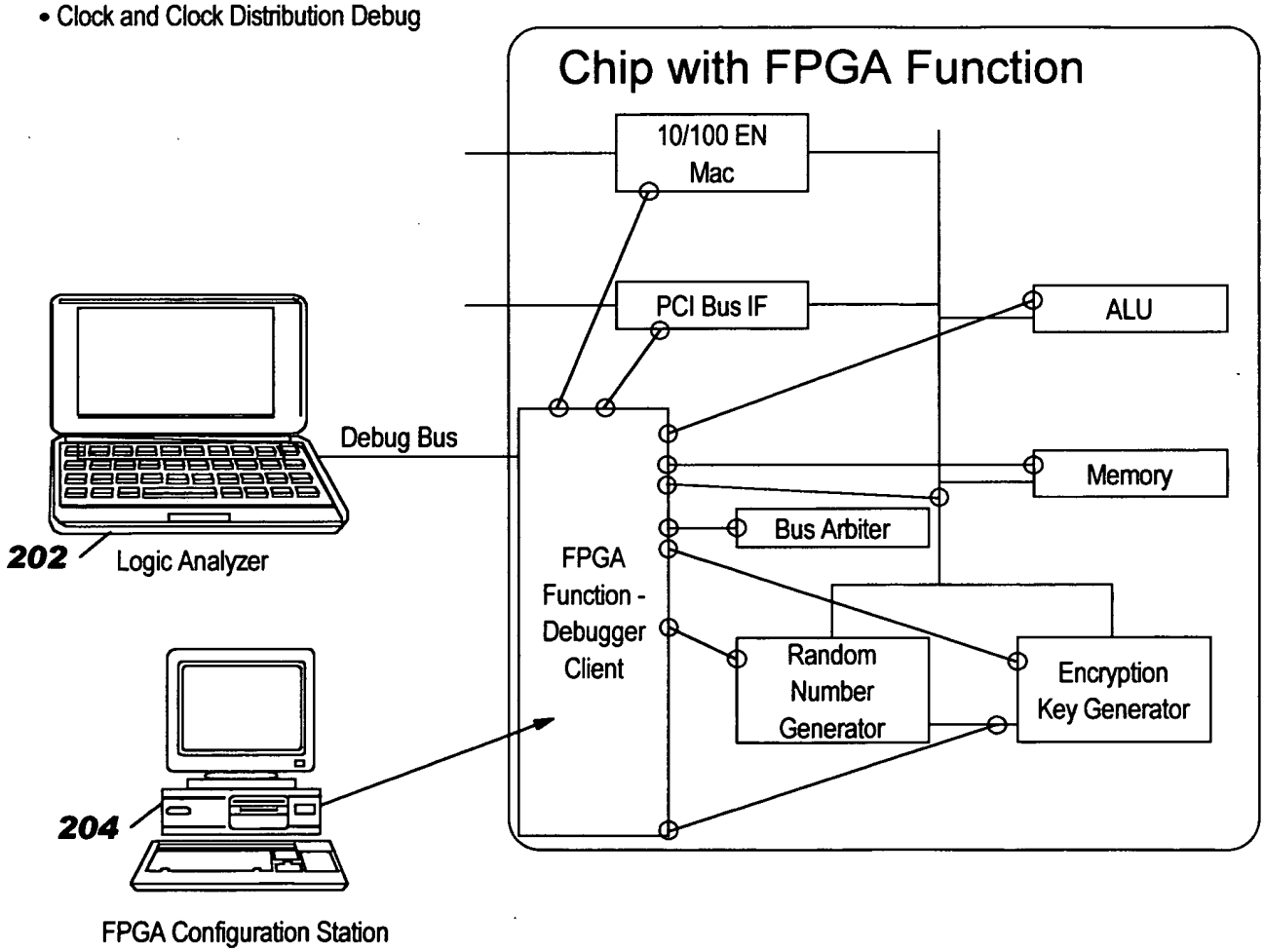

Chip with FPGA Functions
To internal chip bus and signals
External Chip I/O
External Communication Logic
Signal and Bus Selector Logic
122
124
FPGA Internal to External Signal Connector
120

# FIG. 3


External Chip I/O
Monitoring internal bus reveals Function A is not sending correct values to Function B.
Drive correct values into Function B from FPGA and monitor output of Function B to determine if it is operating correctly
External Chip I/Os
130
Function A
132
Function B
FPGA Function
134
Function C
120

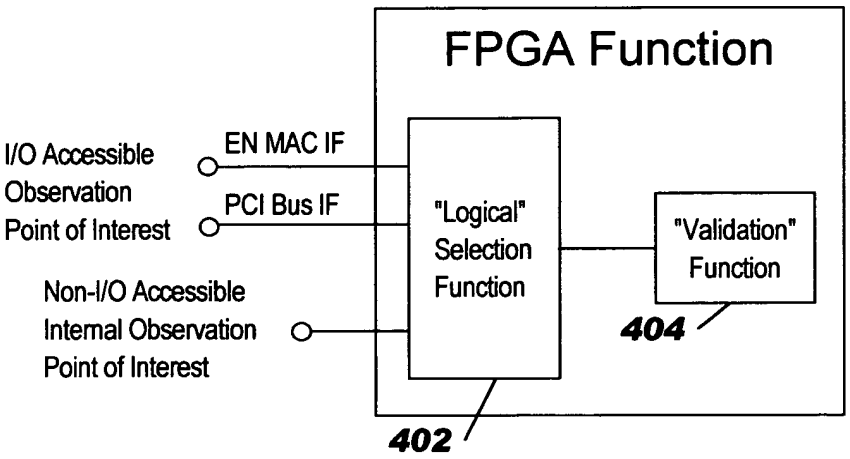
# FIG. 4

- Memory and Memory Bus Debug
- MAC Bus Debug
- PCI, Internal Bus and Bus Arbiter Debug
- Processor Debug
- Clock and Clock Distribution Debug

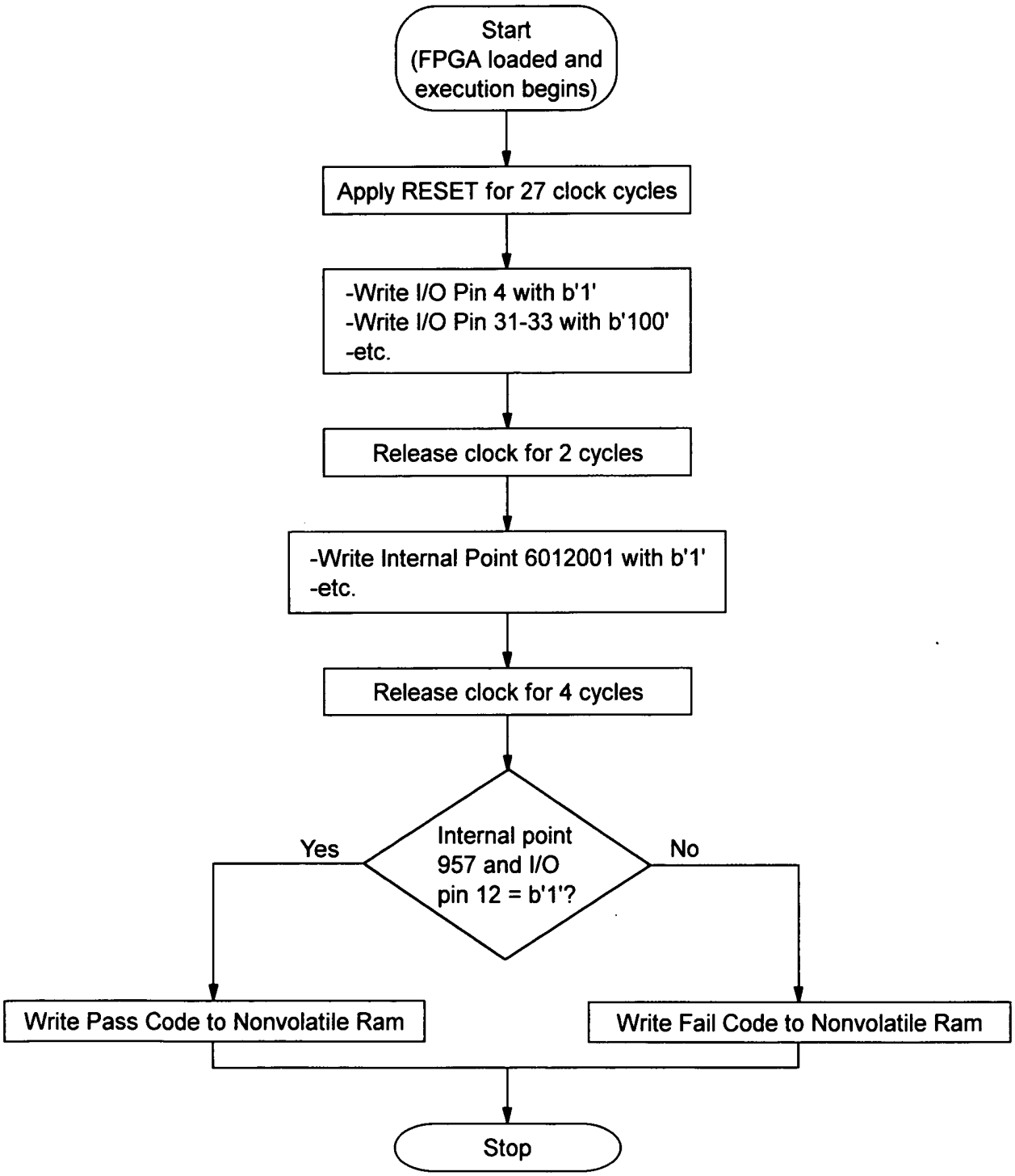



200

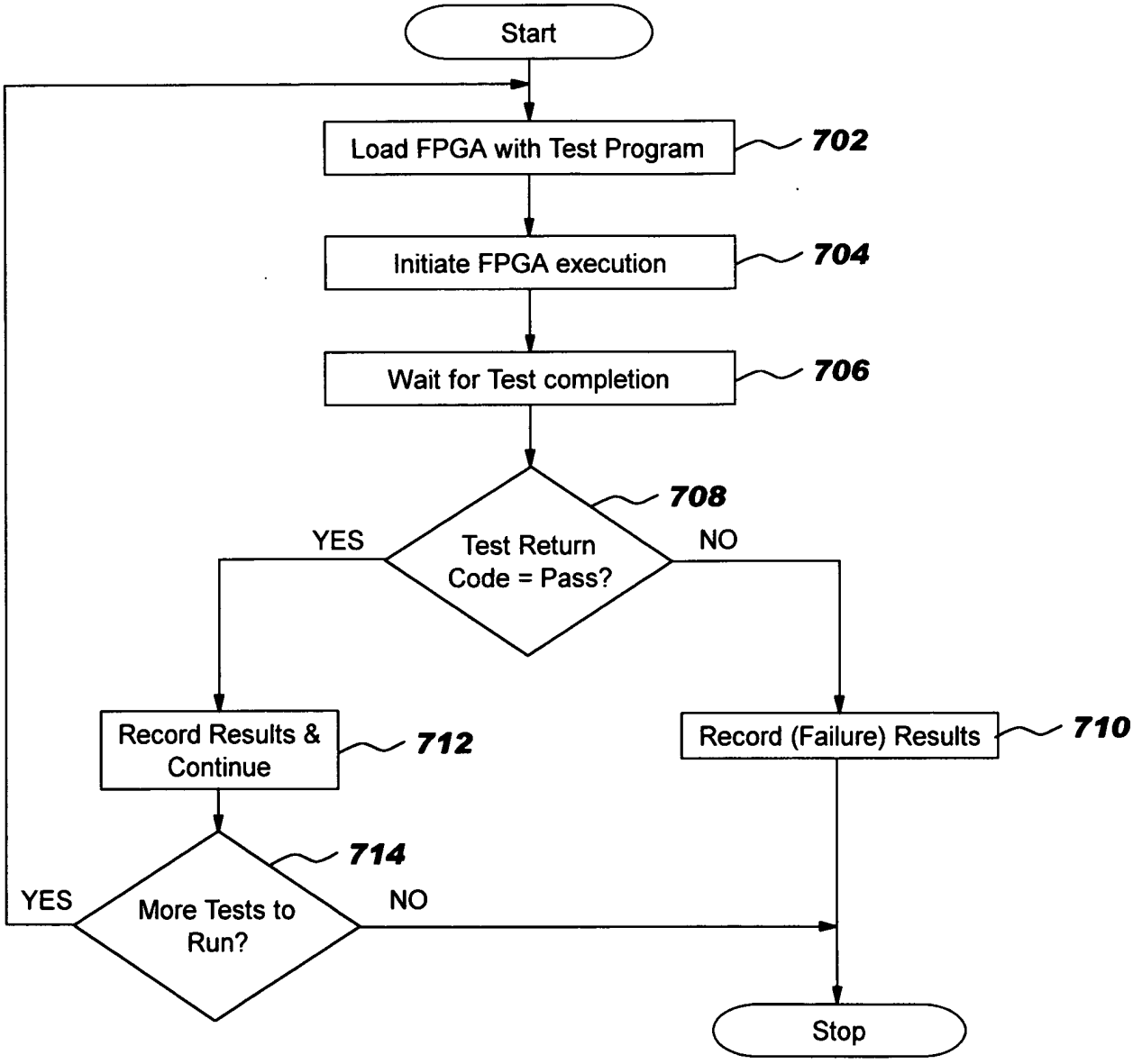
FIG. 5


FPGA Function
I/O Accessible
Observation
Point of Interest
EN MAC IF
PCI Bus IF
Non-I/O Accessible
Internal Observation
Point of Interest
"Logical"
Selection
Function
"Validation"
Function
404
402

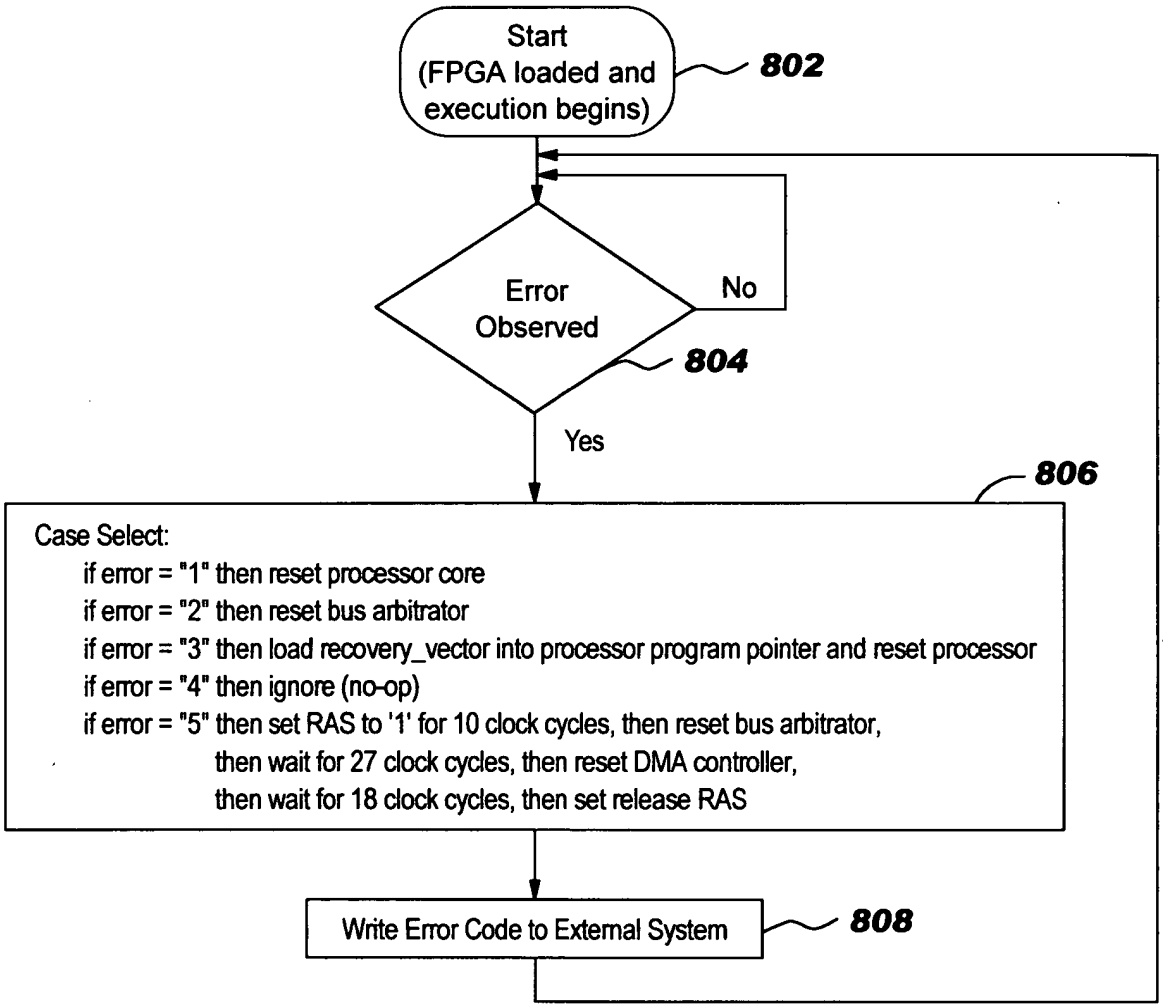
# FIG. 6


Start
(FPGA loaded and
execution begins)
Apply RESET for 27 clock cycles
-Write I/O Pin 4 with b'1'
-Write I/O Pin 31-33 with b'100'
-etc.
Release clock for 2 cycles
-Write Internal Point 6012001 with b'1'
-etc.
Release clock for 4 cycles
Internal point
957 and I/O
pin 12 = b'1'?
Yes
No
Write Pass Code to Nonvolatile Ram
Write Fail Code to Nonvolatile Ram
Stop

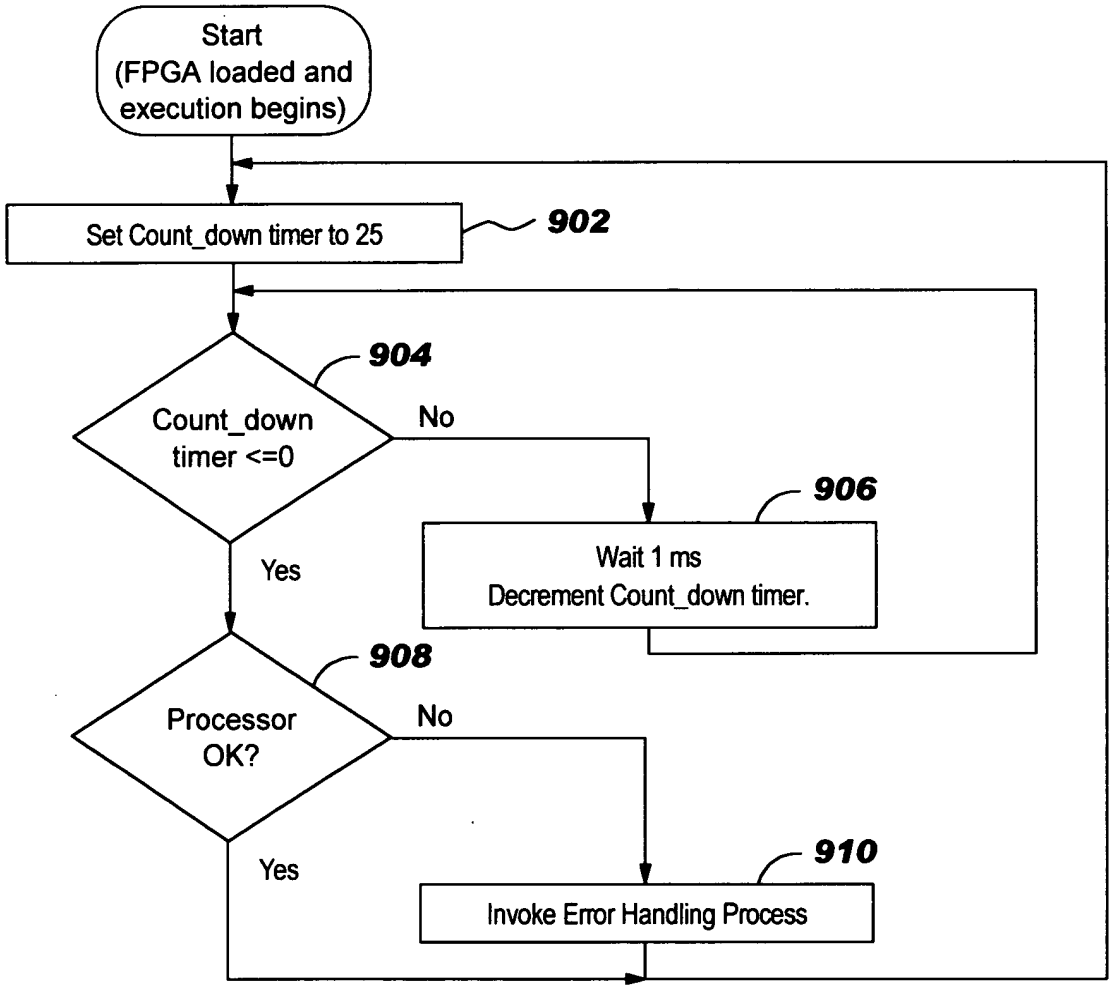
# FIG. 7


Start
Load FPGA with Test Program
702
Initiate FPGA execution
704
Wait for Test completion
706
708
Test Return Code = Pass?
YES
NO
Record Results & Continue
712
Record (Failure) Results
710
714
More Tests to Run?
YES
NO
Stop

# FIG. 8


Start
(FPGA loaded and
execution begins)
802
Error
Observed
No
804
Yes
806
Case Select:
if error = "1" then reset processor core
if error = "2" then reset bus arbitrator
if error = "3" then load recovery_vector into processor program pointer and reset processor
if error = "4" then ignore (no-op)
if error = "5" then set RAS to '1' for 10 clock cycles, then reset bus arbitrator,
then wait for 27 clock cycles, then reset DMA controller,
then wait for 18 clock cycles, then set release RAS
Write Error Code to External System
808

# FIG. 9


Start
(FPGA loaded and
execution begins)
Set Count_down timer to 25
902
904
Count_down
timer <=0
No
Yes
906
Wait 1 ms
Decrement Count_down timer.
908
Processor
OK?
No
Yes
910
Invoke Error Handling Process